

Cours d'Algorithmique

Florent Hivert

Mél : Florent.Hivert@lri.fr

Adresse universelle : <http://www.lri.fr/~hivert>

Informations pratiques

- 25 heures de cours ;
- 25 heures de travaux dirigés ;
- Évaluation :
2 contrôles + examen final + mini projets.

Plan du cours

- 1 Introduction ;
- 2 Notion de complexité ;
- 3 Tableaux et type de donnée abstraits ;
- 4 Listes chaînées ;
- 5 Trouver et trier ;
- 6 Arbres ;

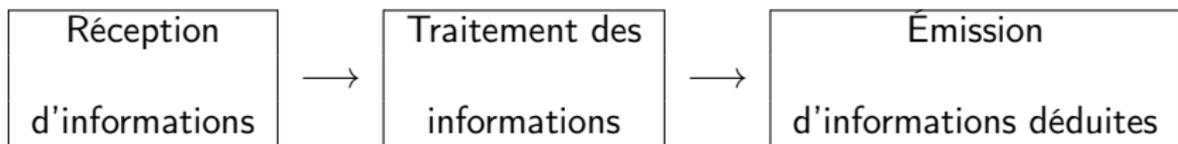
Généralités sur le traitement de l'information

Les ordinateurs sont utilisés pour

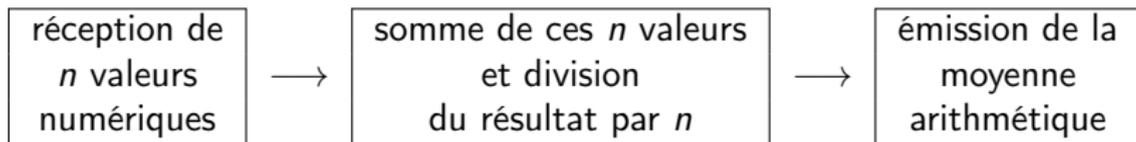
- le traitement d'informations ;
- le stockage d'informations.

Généralités sur le traitement de l'information (2)

Le schéma global d'une application informatique est toujours le même :



Exemple :



La notion de programme

Tout traitement demandé à la machine, par l'utilisateur, est effectué par l'exécution séquencée d'opérations appelées **instructions**. Une suite d'instructions est appelée un **programme**.

Retenir

*Un programme est une **suite d'instructions** permettant à une système informatique d'exécuter une tâche donnée*

écrit dans un langage de programmation compréhensible (directement ou indirectement) par un ordinateur.



Qu'est-ce que la programmation ?

Retenir

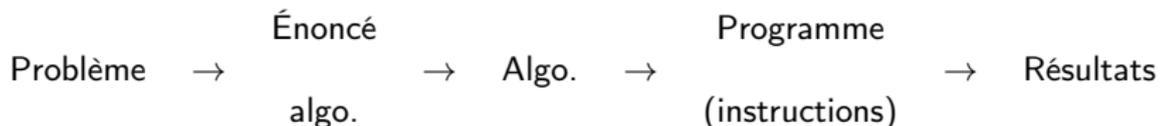
Écriture d'un **programme** dans un langage de programmation donnée.

Langage de programmation : ensemble de règles de vocabulaire et de grammaire compréhensible par un ordinateur.

```
int fun(int x) {
    x = x + 1;
    return x;
}
void main(void) {
    printf("%i", fun(2));
}
```

La notion d'algorithme

Pour résoudre un problème de manière informatique il y a un cheminement méthodique à respecter :



La notion d'algorithme (2)

Définition (Notion d'Algorithme)

- *description formelle d'un **procédé de traitement** qui permet, à partir d'un ensemble d'**informations initiales**, d'obtenir des **informations déduites** ;*
- *succession finie et non ambiguë d'opérations ;*
- *se termine toujours (Note : semi-algorithme).*

Définition (Notion de Programme)

- *suite d'instructions définies dans un langage donné ;*
- *décrit un algorithme.*

LA CUISINE POUR TOUS (G. Mathiot - Le Livre de Poche)**484  Émincé de rouget au pistou**

24 heures à l'avance
Préparation : 40 mn - Cuisson : 30 à 35 mn

Écailler les rougets, les laver, puis soulever délicatement les filets et enlever les arêtes qui restent avec une pince à épiler. Mettre les filets dans un plat creux ; arroser avec de l'huile d'olive ; saupoudrer d'herbes de Provence et laisser mariner au frais, pendant 24 heures, le tout couvert par un torchon.

Cuire le fenouil à l'eau bouillante salée, citronnée et parfumée à l'huile d'olive et avec une brindille de thym (15 à 20 mn). Égoutter.

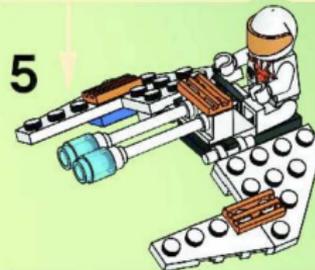
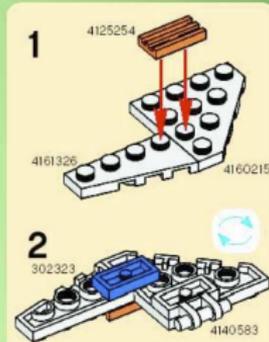
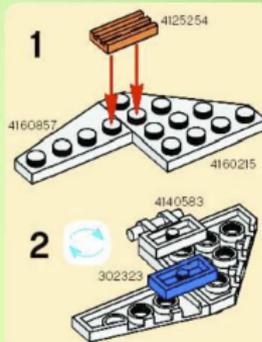
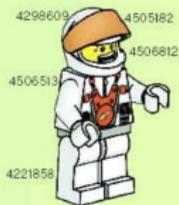
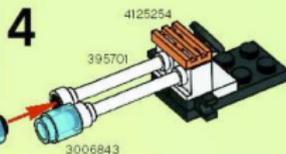
Cuire les pâtes à l'eau bouillante salée, huilée, pendant 12 mn. Égoutter. Rafraîchir. Tenir au chaud.

Pendant ces cuissons préparer le coulis de tomates (41).

Assaisonner le fenouil coupé en tranches avec la vinaigrette. Tenir au chaud. Assaisonner les pâtes avec le basilic (à volonté) haché et un mélange d'huile d'olive et de vinaigre de xérès. Disposer les pâtes sur le plat, recouvrir avec la fondue de fenouil.

Rapidement, cuire à la poêle Tefal, à feu très vif, les filets marinés pour qu'ils deviennent dorés et croustillants. Saler. Poirver et disposer en étoile les filets de rougets, sur le plat de légumes. Servir avec le coulis de tomates en saucière.

1 kg 500 rougets.
400 g pâtes.
500 g fenouil.
2 dl huile olive.
0 dl 5 vinaigre xérès.
Coulis de tomates.
1 citron.
Basilic.
Herbes de Provence.
Thym.
Vinaigrette.
Sel.
Poivre.



⚠ **AVERTISSEMENT : RISQUE D'ÉTOUFFEMENT.**
Contient des petites pièces. Ne convient pas aux enfants de moins de 3 ans.

⚠ **ADVERTENCIA: PELIGRO DE ASFIXIA.**
No recomendado para niños menores de 3 años.
Contiene piezas pequeñas.



Exemple d'algorithme

Algorithme (Calcul de la moyenne d'un tableau de notes)

- *Entrées : un entier n , un tableau de n nombres à virgules*
- *Sorties : la moyenne des nombres du tableau*

i est un entier ; s un nombre à virgule.

initialiser s avec 0

pour i de 1 à n faire

ajouter le i -ème élément du tableau à s

retourner s/n

Exemple d'algorithme

Algorithme (Calcul de la moyenne d'un tableau de notes)

- *Entrées* : un entier n , un tableau de n nombres à virgules
- *Sorties* : la moyenne des nombres du tableau

i est un entier ; s un nombre à virgule.

initialiser s avec 0

pour i de 1 à n faire

ajouter le i -ème élément du tableau à s

retourner s/n

Quel est le nombre d'additions effectuées par cet algorithme ?

Différentes problématiques

- **terminaison** : terminera en un temps fini.
- **complexité en temps** : terminera en un temps borné (raisonnable).
- **complexité en espace** : terminera en utilisant une quantité de mémoire bornée (raisonnable).
- **correction** : si l'algorithme termine en donnant une proposition de solution, alors cette solution est correcte.
- **complétude** : pour un espace de problèmes donné, l'algorithme, s'il termine, donnera toujours des propositions de solutions.

Pourquoi faire

“ I will, in fact, claim that the difference between a bad programmer and a good one is whether he considers his code or his data structures more important. Bad programmers worry about the code. Good programmers worry about data structures and their relationships. ” — Linus Torvalds (creator of Linux)

“ Algorithms + Data Structures = Programs. ” — Niklaus Wirth

Objectifs du cours d'algorithmique

- Concevoir des algorithmes simples ;
- Analyser les performances d'un algorithme :
notion de complexité ;
- Algorithmes fondamentaux : description et complexité ;
- Structures de données performantes :
tableaux dynamiques, tableaux triés, listes chaînées, arbres,
tables de hachage.

Choisir la bonne structure de donnée selon l'application !!!

Objectifs du cours d'algorithmique

- Concevoir des algorithmes simples ;
- Analyser les performances d'un algorithme :
notion de complexité ;
- Algorithmes fondamentaux : description et complexité ;
- Structures de données performantes :
tableaux dynamiques, tableaux triés, listes chaînées, arbres,
tables de hachage.

Choisir la bonne structure de donnée selon l'application !!!

Bibliographie

Mots clés : algorithmique, analyse d'algorithmes.

-  Cormen, Leiserson, Rivest et Stein, *Algorithmique : Cours avec 957 exercices et 158 problèmes*, (trad. de l'anglais) Dunod, 2010.
-  Knuth, *The Art of Computer Programming (TAOCP)* Vol 1.-5., Addison-Wesley, Boston.
-  Graham, Knuth et Patashnik, *Mathématiques concrètes, Fondations pour l'Informatique*, International Thomson publishing France, 1998.
-  Froidevaux, Gaudel et Soria, *Types de données et algorithmes* McGraw-Hill 1990– Ediscience International 1993, 577 pages
-  Granjon, *Algorithmes en Pascal et en langage C*, Dunod, 2004.